

thingsboard基本功能展示及步骤

添加资产和设备，并连接其关系

新建资产，并连接资产之间关系

添加传感器设备

在Building A资产中关联此传感器设备。

增加资产的属性，如地理位置

创建仪表盘，并添加地图部件

添加一个dashboards，进行编辑。

在实体别名中添加别名

添加实体部件

在不同的小部件中创建动作以在状态之间导航

添加和配置新的仪表板状态

通过代码进行数据的发送

在进行数据的发送之前，先对数据发送方式做个了解和测试

客户端API

服务端API

设备添加遥测

添加设备列表在实体表

增加地图部件

在设备中添加属性实现在上一步中添加的图片中看到传感器具体位置

在地图上对资产和设备添加链接

编辑仪表盘中default中的地图，添加actions

添加新的仪表盘状态

添加实体的多个action

添加传感器展示部件

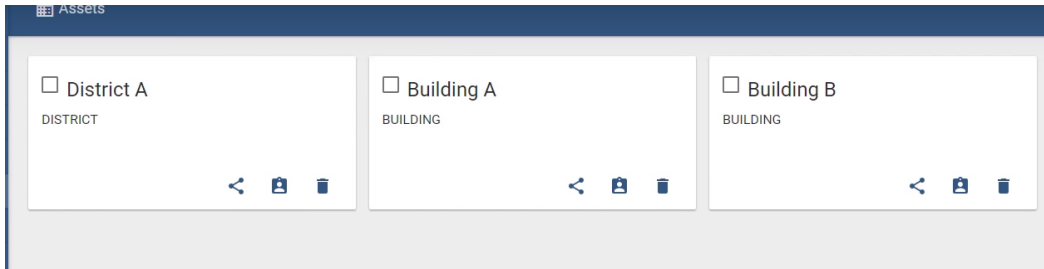
thingsboard基本功能展示及步骤

参考资料

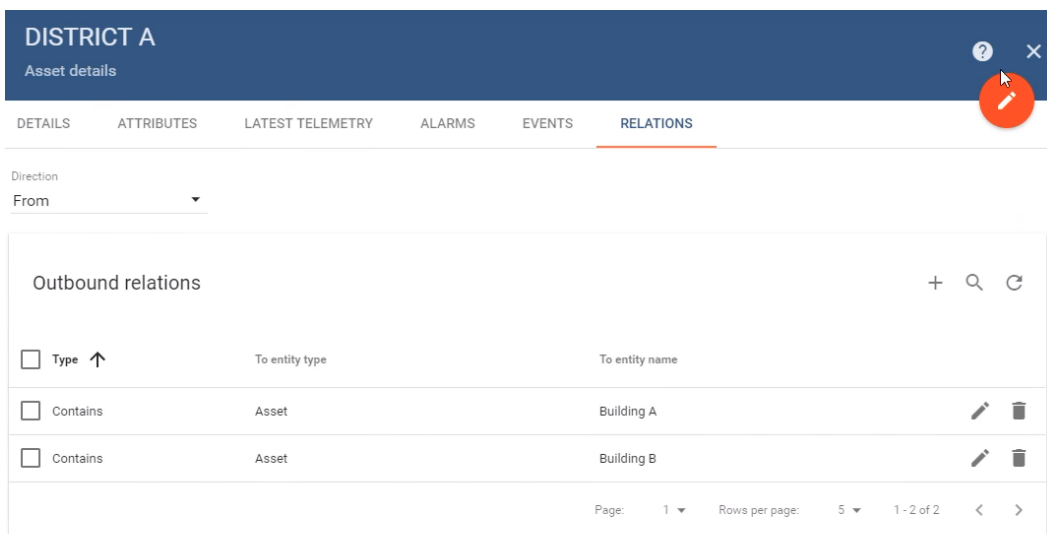
添加资产和设备，并连接其关系

新建资产，并连接资产之间关系

在ASSETS中编辑，添加资产District A，类型设置为district，添加资产 Building A，类型设置为building，添加资产Building B，类型设置为building如下：



编辑District A，Direction选择为from，在RELATIONS中添加relation，选择资产Building A进行绑定，同时将Building B用相同的方法进行绑定。如下：



添加传感器设备

在DEVICES中，编辑，添加传感器设备，例如：温度传感器等

TEMPERATURE SENSOR

Device details


DETAILS


ATTRIBUTES

LATEST TELE

MAKE DEVICE PUBLIC

ASSIGN TO CUSTOMER

 COPY DEVICE ID

 COPY ACCESS TOKEN

Name *

temperature sensor

Device type*

temperature

☐ Is gateway

Description

在Building A资产中关联此传感器设备。

编辑Building A， Direction选择为from， 在RELATIONS中添加relation， 类型为：Contains， 选择设备， 对该温度传感器进行绑定， 如有更多传感器用相同的方法进行绑定。
如下：

Edit relation

×

Relation type *

Contains

To entity

TypeDevice

Device *temperature sensor

Additional info (JSON)

SAVE

CANCEL

增加资产的属性，如地理位置

编辑Building A，在ATTRIBUTE中添加key和value，
key:address，value: (T String类型) 南京市雨花台区软件大道168号润和创智中心C
栋3层
如下：

Add attribute

×

Key *

address

Value type

T String

String value *

南京市雨花台区软件大道168号润

ADD

CANCEL

添加经度

key:latitude，value: (123 Double类型) 118.764288


添加纬度

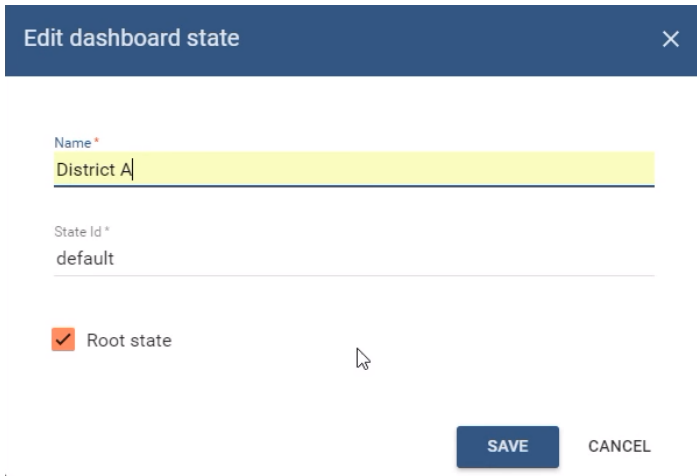
key:longitude，value: (123 Double类型) 31.979344

同理，资产Building B如上，添加具体位置属性。

创建仪表盘，并添加地图部件

添加一个dashboards，进行编辑。

点击仪表盘状态管理，编辑状态，将状态名设置为资产District A名，并为根状态。

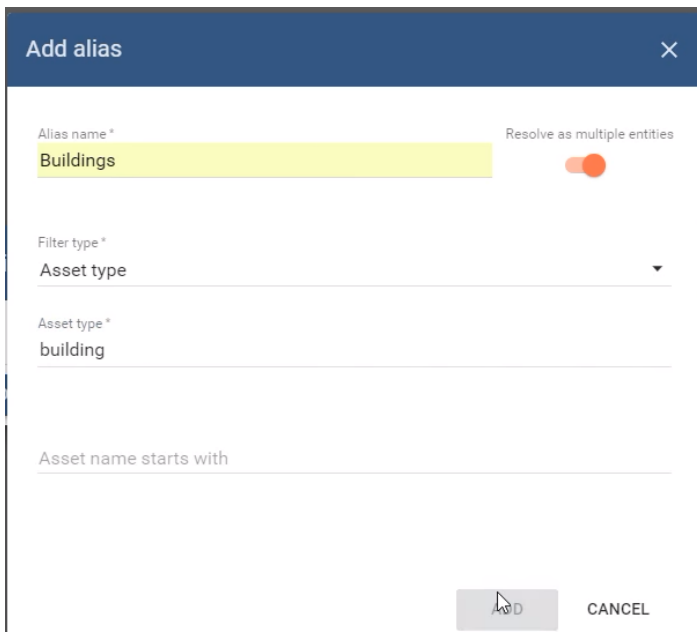


The 'Edit dashboard state' dialog box contains the following fields and controls:

- Name ***: A text input field containing 'District A'.
- State Id ***: A text input field containing 'default'.
- Root state**: A checkbox that is checked.
- Buttons**: 'SAVE' and 'CANCEL' buttons at the bottom right.

在实体别名中添加别名

筛选类型选择资产，资产类型选择创建资产的类型building，保存。



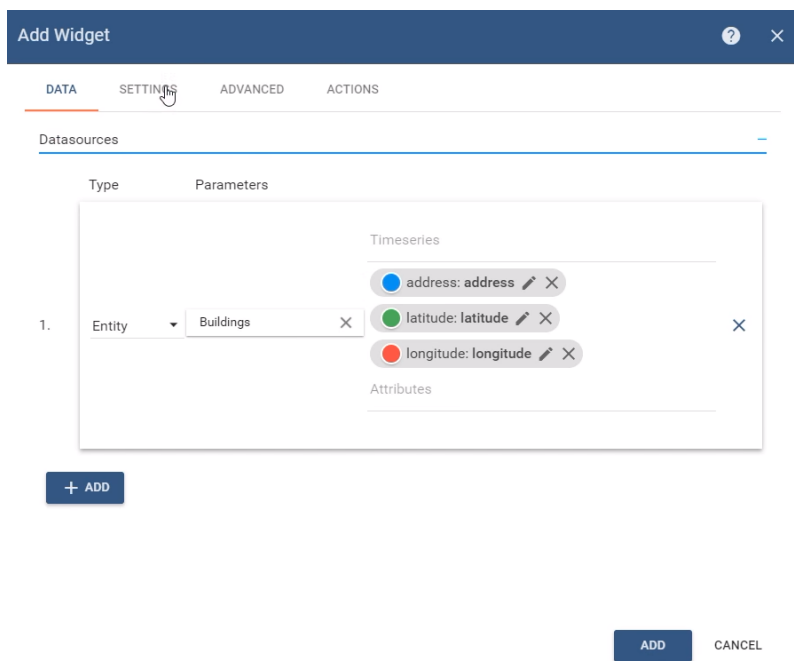
The 'Add alias' dialog box contains the following fields and controls:

- Alias name ***: A text input field containing 'Buildings'.
- Resolve as multiple entities**: A toggle switch that is turned on.
- Filter type ***: A dropdown menu with 'Asset type' selected.
- Asset type ***: A text input field containing 'building'.
- Asset name starts with**: An empty text input field.
- Buttons**: 'ADD' and 'CANCEL' buttons at the bottom right.

添加部件

选择部件包为Maps，LATEST VALUES（最新值）中的OpenStreetMap地图，在data中添加数据源，类型为实体，实体别名选择上一步添加的实体Buildings，属性中选择资产Buildings A中设置的具体地理位置属性。

如下：



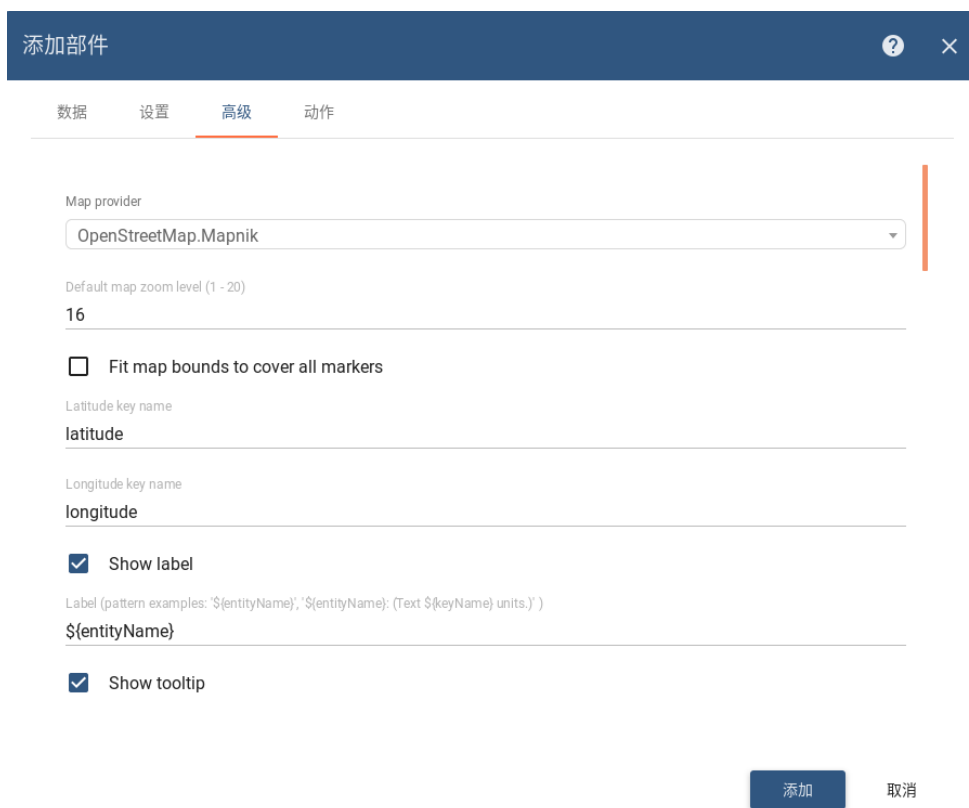
设置中，修改标题为自己部件图的标题

高级中，将Map provider选择为OpenStreetMap.HOT

level: 16

fit map 去掉勾选

如下：



Tooltip样式去除以下，只留下显示经纬度属性。

添加部件

数据

设置

高级

动作

☒ Auto-close tooltips

Tooltip (for ex. 'Text \${keyName} units.' or <link-act name='my-action'>Link text</link-act>)
\${entityName}

Latitude: \${latitude:7}
Longitude: \${longitude:7}

Color

rgb(254, 117, 105)

×

☒ Use color function

Color function: f(data, dsData, dsIndex)

```
i 1 var type = dsData[dsIndex]['type'];
2 if (type == 'colorpin') {
3   var temperature = dsData[dsIndex]['temperature'];
4   if (typeof temperature !== undefined) {
5     var percent = (temperature + 60)/120 * 100;
6     return tinycolor.mix('blue', 'red', amount = percent).toHexString();
7   }
8   return 'blue';
9 }
```

JAVASCRIPT TIDY

Custom marker image

No image

Drop an image or click to select a file to upload

×

添加

取消

添加保存即可在仪表盘上看到资产的具体地理位置，点击地图上资产可以显示具体的坐标属性。

添加实体部件

编辑添加部件，选择Cards，选择LATEST VALUES（最新值）中的Entities，在data中添加数据源，类型为实体，实体别名选择上一步添加的实Buildings，属性中选择资产Buildings A中设置的address属性，并编辑属性，将标签首字母大写，以区分key和标签。保存

数据键配置

×

设置

高级

键 *

address

标签 *

Address

颜色

#2196f3

单位符号

小数位数

☐

使用数据后处理功能

保存

取消

高级中，加入实体表title
去除Display entity type column选项。添加并保存。
如下：

Add Widget

?

×

DATA

SETTINGS

ADVANCED

ACTIONS

Entities table title

Buildings list

☒

Enable entities search

☒

Display entity name column

Entity name column title

☐

Display entity type column

☒

Display pagination

Default page size

10

Default sort order

entityName

ADD

CANCEL

添加保存。

在不同的小部件中创建动作以在状态之间导航

添加和配置新的仪表板状态

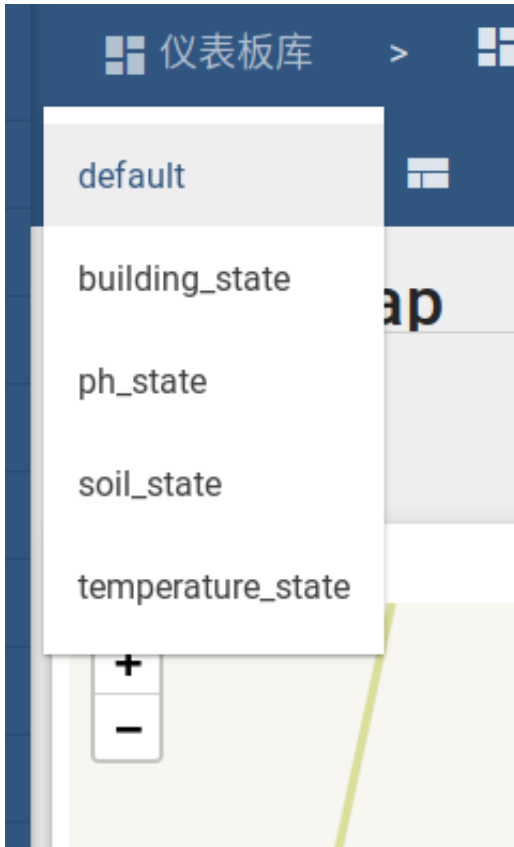
编辑仪表盘状态，添加仪表盘状态

状态名为: \${entityName}

ID: 自定义

保存，如下：

可以看到在此仪表盘下添加的多个dashboard



在default的dashboard，编辑实体部件，在ACTIONS中添加动作

动作源：On row click（点击行）

Type：Navigate to new dashboard state（切换到新仪表盘状态）

选择目标仪表盘状态：新建的仪表盘状态

编辑动作

✕

动作源 *

点击行

动作名称 *

row_click

图标

more_horiz

类型 *

切换到新仪表板状态

目标仪表板状态 *

building_state

打开右侧布局 (移动端视图)

☐

☒ 从部件中设置实体

状态实体参数名称

默认

保存

取消

保存

通过代码进行数据的发送

在进行数据的发送之前，先对数据发送方式做个了解和测试

基于mqtt对数据进行发送到thingsboard平台

[mqtt_api](#)

传送数据格式

```
1 # Key始终是一个字符串，而value可以是string, boolean, double或long。
2 # 也可以使用自定义二进制格式或某些序列化框架。
3 {"stringKey":"value1", "booleanKey":true, "doubleKey":42.0,
4  "longKey":73}
```

客户端API

1.遥测数据api

Publish消息的主题及数据格式

```
1 # 主题
2 v1/devices/me/telemetry
3 # 数据格式
4 {"key1":"value1", "key2":"value2"}
5 [{"key1":"value1"}, {"key2":"value2"}]
```

1. 属性api

(1) 将属性更新发布到服务器

Publish消息的主题及数据格式

```
1 # 主题
2 v1/devices/me/attributes
3 # 数据格式
4 {"attribute1":"value1", "attribute2":true, "attribute3":42.0,
  "attribute4":73}
```

(2) 从服务器请求属性值

Publish消息的主题及数据格式

```
1 # 主题
2 v1/devices/me/attributes/request/$request_id
3 # 其中$ request_id是整数请求标识符。在发送带有请求的PUBLISH消息之前，
  客户端需要订阅如下主题
4 v1/devices/me/attributes/response/+
5 # 数据格式
6 {"key1":"value1"}
```

服务端API

Publish消息的主题及数据格式

```
1 # 从服务器订阅RPC命令，请将SUBSCRIBE消息发送到以下主题：
2 v1/devices/me/rpc/request/+
3 # 订阅后，客户端将收到单个命令作为相应主题的PUBLISH消息：
4 v1/devices/me/rpc/request/$request_id
5 # 其中$ request_id是整数请求标识符。
6 # 客户端应将响应发布到以下主题：
7 v1/devices/me/rpc/response/$request_id
```

1. 连接平台返回数据是否成功状态列表：

```
1 0x00已连接 - 已成功连接到ThingsBoard MQTT服务器。
2 0x04连接被拒绝，用户名或密码错误 - 用户名为空。
3 0x05连接被拒绝，未经授权 - 用户名包含无效的$ ACCESS_TOKEN。
4
```

设备添加遥测

进入需要遥测的设备中，复制设备的访问令牌，并添加到代码中
代码如下：

```
1 import paho.mqtt.publish as publish
2 import time
3 import json
4 import random
5
6 HOST = "192.168.3.89"
7 PORT = 1883
8 username = "PnnLdrZ5GdsXnXkILM6R"
9 topic = "v1/devices/me/telemetry"
10
11 if __name__ == '__main__':
12     client_id = time.strftime('%Y%m%d%H%M%S',
13                               time.localtime(time.time()))
14
15     while True:
16         i = random.randint(20, 25)
```

```

16     data = {"temperature": i}
17     time.sleep(1)
18     print (data)
19     publish.single(topic, json.dumps(data), qos=1,
hostname=HOST, port=PORT, client_id=client_id,
20         auth={'username': username, 'password':
"public"})

```

添加设备列表在实体表

1. 编辑之前创建的dashboard，添加实体别名

筛选类型为：Device search query（设备搜索查询）

选作为根实体

关联类型：Contains

设备类型将之前所需要的已创建设备添加进去

如下：

The screenshot shows the 'Add alias' dialog box with the following configuration:

- Alias name ***: Building Sensors
- Resolve as multiple entities**: ☒
- Filter type ***: Device search query
- Root entity**: ☒ Use dashboard state entity as root
- State entity parameter n...**: By default
- Direction ***: From
- Max relation level**: 1
- Relation type**: Contains
- Device types ***: energy sensor

A dropdown menu is open for 'Device types', showing options: default, energy sensor, temperature sensor, and water sensor. The 'ADD' button is highlighted.

添加，保存。

2. 在dashboard中切换到新加的dashboard（building_state）中，添加部件
添加Cards中的entity部件，选择上一步建的实体别名，

在时间序列中添加传感器key，即要展示的数据实体

Add Widget

DATA

SETTINGS

ADVANCED

ACTIONS

Datasources

Maximum 1 datasource is allowed.

Type

Parameters

1.

Entity

Building Sensors

energy: energy

temperature: temperature

water: water

Timeseries

Attributes

ADD

CANCEL

编辑这些时间序列，
标签首字母大写
使用数据后处理功能，将数据添加单位

```
1 return value? value + ' °C' : '-';
2
```

数据键配置

设置

高级

键 *

temperature

标签 *

Temperature

颜色

#f44336

单位符号

小数位数

☒ 使用数据后处理功能

function (time, value, prevValue) {

1 return value? value + ' °C' : '-';

整洁

保存

取消

在高级中
命名，并去除实体类型显示这一栏

数据

设置

高级

动作

Entities table title

Sensor List

☒

Enable entities search

☒

Display entity name column

Entity name column title

☐

Display entity type column

☒

Display pagination

Default page size

10

Default sort order

entityName

添加，保存，即可看到遥测数据。如下：

Sensor List

实体名 ↑	Soil	Ph	Temperature
ph sensor	-	7	-
soil sensor	48%		-
temperature sensor	-		35.75 °C

Page: 1

Rows per page: 10

1 - 3 of 3

增加地图部件

在资产Building A中增加属性

key: imageUrl

String Value: (T String) (可添加静态图片地址，或者视频)

如: <http://img0.imgtn.bdimg.com/it/u=2992282432,2620339158&fm=26&gp=0.jpg>

返回仪表盘，进入building_state，

添加别名，命名，选择实体（仪表盘状态）

Add alias [X]

Alias name *
Selected Building

Resolve as multiple entities ☐

Filter type *
Entity from dashboard state

State entity parameter name
By default

Default state entity
Type

[ADD] [CANCEL]

添加，保存。

添加部件

选择Maps中的Image Map

添加数据源

实体选择

在属性中添加xPos,yPos

Add Widget [?] [X]

DATA SETTINGS ADVANCED ACTIONS

Datasources

Type	Parameters
1. Entity	Building Sensors Attributes: xPos: xPos, yPos: yPos

[+ ADD]

[ADD] [CANCEL]

在设置中给标题命名

在高级中，将原背景清除

Image URL source entity alias:

Image URL source entity attribute:imageUrl

tooltip: 去除多余样式

保存，可以看到自己添加的图片

在设备中添加属性实现在上一步中添加的图片中看到传感器具体位置

选择服务端属性 (Server attributes)

添加

key: xPos

String Value: (123 Double) 0.515

key: yPos

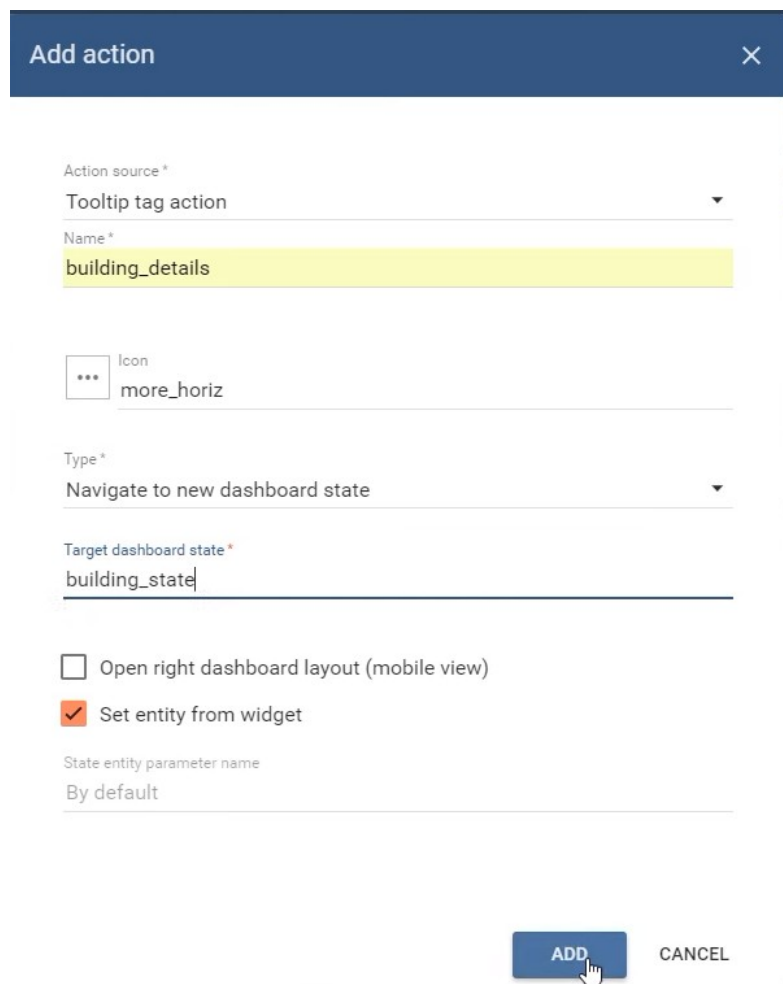
String Value: (123 Double) 0.315

在地图上对资产和设备添加链接

编辑仪表盘中default中的地图，添加actions

选择: tooltip tag action, 命名, 类型: 跳转到新的仪表盘状态, 选择除了default之外的

新的仪表盘状态。



Add action

Action source *

Tooltip tag action

Name *

building_details

Icon

more_horiz

Type *

Navigate to new dashboard state

Target dashboard state *

building_state

☐ Open right dashboard layout (mobile view)

☒ Set entity from widget

State entity parameter name

By default

ADD CANCEL

在高级中修改样式，添加跳转link

```
1 <link-act name='farm_details'>Farm details</link-act><br/>
```

Tooltip (for ex. 'Text \${keyName} units.' or <link-act name='my-action'>Link text</link-act>)

\${entityName}

Latitude: \${latitude:7}
Longitude: \${longitude:7}

<link-act name='farm_details'>Farm details</link-act>

添加新的仪表盘状态

仪表盘状态编辑

状态名 *

\${entityName}

状态ID *

temperature_state

☐ 根状态

保存

取消

添加实体的多个action

编辑新加仪表盘中的entity， 增加action
点击行（on row click）

编辑动作

动作源 *

点击行

动作名称 *

row_click

...

 图标

more_horiz

类型 *

自定义动作

function (\$event, widgetContext, entityId, entityName, additionalParams) {

整洁

1 var \$injector = widgetContext.\$scope.\$injector;

2 \$injector.get('deviceService').getDevice(entityId.id

3).then(function(device){

4 if(device.type == 'temperature'){

5 open('temperature_state');

6 }else if(device.type == 'soil'){

7 open('soil_state');

8 }else{

9 open('ph_state');

10 };

11 }}

12 }

保存

取消

```

1  var $injector = widgetContext.$scope.$injector;
2  $injector.get('deviceService').getDevice(entityId.id).then(function(device){
3      if(device.type == 'temperature'){
4          open('temperature_state');
5      }else if(device.type == 'soil'){
6          open('soil_state');
7      }else{
8          open('ph_state');
9      }
10 });
11
12 function open(stateId){
13     var param = {
14         entityId:entityId,
15         entityName:entityName
16     };
17     widgetContext.stateController.openState(stateId,param,false);
18 }
19
20
21

```

同理，编辑新加仪表盘中的map，在actions中增加action
(Tooltip tag action)

在ADVANCED高级中在样式中添加传感器详情的链接

添加传感器展示部件

添加传感器实体别名，如下：

Add alias

Alias name *

Selected Energy Meter

Resolve as multiple entities

Filter type *

Entity from dashboard state

State entity parameter name

By default

Default state entity

Type

ADD

CANCEL

进入传感器的仪表盘（dashboard）添加所需要部件

添加数据源，在部件中选择实体，参数选择传感器的实体别名，在时间序列中添加传感器其参数

1. Analogue gauges中，在高级中对最大最小值进行设置，添加标题

2. Charts中，时间坐标可以显示自己设定的时间段

Add Widget

DATA

SETTINGS

ADVANCED

ACTIONS

☐ Use dashboard timewindow

Timewindow

Datasources

PLEASE ADD DATASOURCE

+ ADD

REALTIME - LAST MINUTE

REALTIME

HISTORY

Last

1 minute

Data aggregation function

Average

5 seconds

Grouping interval

5 seconds

UPDATE

CANCEL

ADD

CANCEL

可以配置实体时间序列属性及颜色

Data key configuration

SETTINGS

ADVANCED

Key *

temperature

Label *

#{entityName}: Temperature, °C

Color

#ef5350

Special symbol to show next to value

°C

Number of digits after floating point

2

☐

Use data post-processing function

SAVE

CANCEL

可以在设置中添加显示最大最小值

Number of digits after floating point

LEGEND SETTINGS

Legend position

Bottom

☒ Show min value

☒ Show max value

☒ Show average value

☐ Show total value

ADD

CANCE

可以在高级中添加抬头

Axis title

Temperature, °C

1. Digital gauges部件

可以在高级中设置title，并选中显示

☒

Show gauge title

Unit title

kWh

可以修改元素大小及颜色

